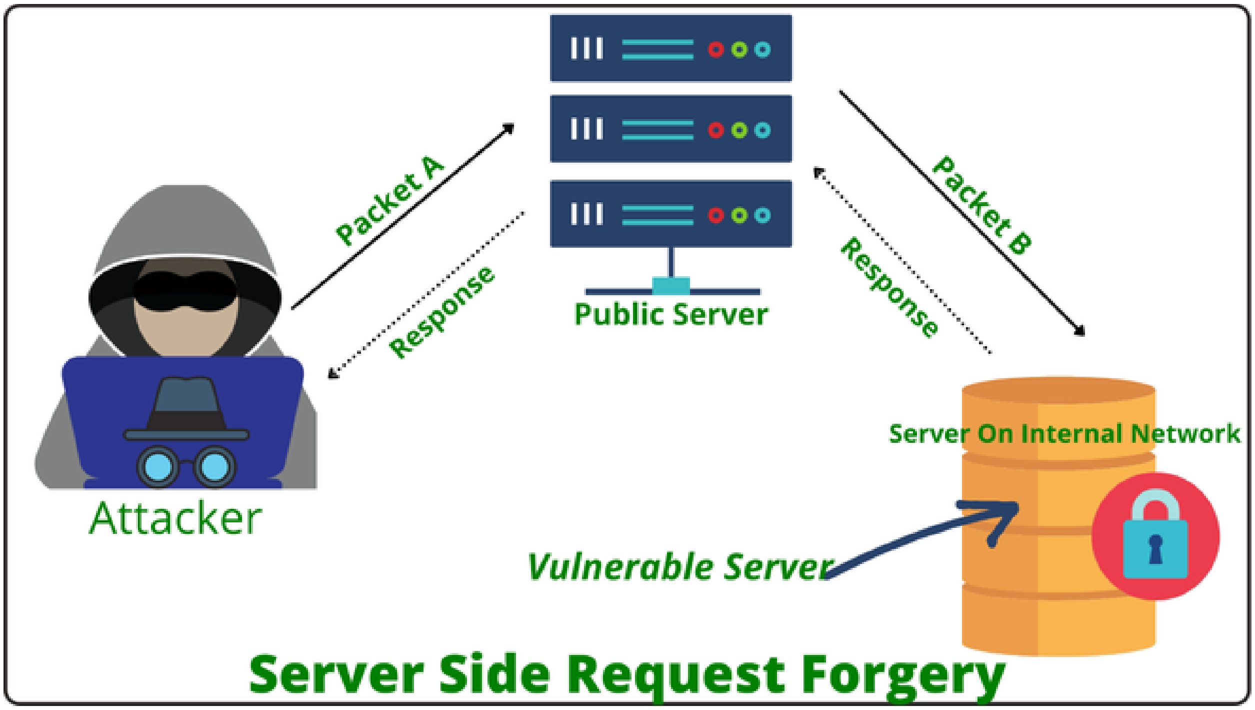
Server Side Request Forgery

What is SSRF?

SSRF stands for Server Side Request Forgery. SSRF is a server site attack that leads to sensitive information disclosure from the back-end server of the application. In server site request forgery attackers send malicious packets to any Internet-facing web server and this web server sends packets to the back end server running on the internal network on behalf of the attacker.



Types

1. **Blind SSRF:** In a Blind SSRF, attackers are not able to control the data of packet B that are sent to the application in a trusted internal network. Here the attacker can control the IP address and ports of the server. To exploit this type of SSRF we have to feed the URL followed by the colon and port number, by observing responses and error messages from the server we can find the open and close ports of the server. We have tried this procedure for the different ports to check their status.
   Example : http://example.com:1337 , http://example.com:9923

2. **Partial SSRF :** In this type of SSRF, we get a limited response from the server like the title of the page or get access to resources but can't see the data. We can control only certain parts of packet B that arrive at the internal application. This type of vulnerability can be used to read local system files such as /etc/config, /etc/hosts, etc/passwd, and many others. By using the file:// protocol we can read files on the system. In some cases, XXE injection, DDos, these types of vulnerability may exploit Partial SSRF Vulnerability.
   Example: file:///etc/hosts, file:///etc/config

3. **Full Response SSRF :** In Full SSRF we have complete control over Packet B. Now we can access the services running on the internal network and find the vulnerabilities in the internal network. In this type of SSRF we can use the protocols like file://, dict://, http://, gopher://, etc. here we have a large scope of creating different requests and exploiting the internal network if any vulnerabilities are present. Full SSRF vulnerability may cause the application to crash through a buffer overflow, by sending a large string in the request causes the buffer overflow.

Impact

A successful SSRF attack can often result in unauthorized actions or access to data within the organization, either in the vulnerable application itself or on other back-end systems that the application can communicate with. In some situations, the SSRF vulnerability might allow an attacker to perform arbitrary command execution.

An SSRF exploit that causes connections to external third-party systems might result in malicious onward attacks that appear to originate from the organization hosting the vulnerable application.

Mitigations

- **Whitelisting**: Server only allows a few domain names to be used in the request, the server has a white list of the domain if the domain name from that list matches with a domain name from the request then only accepts the request otherwise server declines the request.

- **Blacklisting**:-Server discards all the requests containing IP addresses, domain names, and keywords from the blacklist of the server.

- **Restricted content**:- Server allows access to only a particular amount of files to the user, it allows only a few file extension types for public access.

References

https://portswigger.net/web-security/ssrf

https://www.acunetix.com/blog/articles/server-side-request-forgery-vulnerability/

https://owasp.org/www-community/attacks/Server_Side_Request_Forgery

https://www.imperva.com/learn/application-security/server-side-request-forgery-ssrf/

https://www.geeksforgeeks.org/server-side-request-forgery-ssrf-in-depth/